A Performance Model for Load Test Components

Máté J. Csorba and Sándor Palugyai

Performance issues of telecommunication system testing, load testing with Testing and Test Control Notation version 3 (TTCN-3) specifically, is investigated in this work. To be able to design test components that simulate behavior of real nodes in a telecom test network and that are even interchangeable with a real node in terms of performance, modeling is desirable. The aim of successful traffic mix composition is to simulate behavior of real nodes, or even to produce an equivalent counterpart of the real node in TTCN-3.

In TTCN-3, configuration of the test system can be set dynamically. This means that a configuration may consist of several components participating in the test, either as a component that communicates directly with the System Under Test (SUT), or as a component having only registration or internal purposes, meaning that it communicates only with parts of the test system itself. The various components used by a test system are called Parallel Test Components (PTCs). PTCs communicate with each other via test ports. Similarly, communication towards the SUT is established via test ports too.

Each test port that connects either two test components (internal port) or a test component and the interface towards the SUT is modeled as a FIFO queue for the incoming/outgoing messages. Properties of the FIFO queues assigned to a test port are dependent on the actual implementation of the TTCN-3 compiler. The queues can be infinite in principle, as long as the system memory lasts, but might overflow indeed. More importantly, in a load test system response time must be considerably short. This means that it is inexpedient to implement a virtually infinite buffer for a PCO and forget about message loss at all. Although a sufficiently long buffer might eliminate message loss, but response time increases significantly at the same time.

The actual behavior of a test case is defined by dynamic behavioral statements in a test component that communicates over certain test ports. Usually, sequences of statements can be expressed as a tree of execution paths called alternatives. In TTCN-3, the alt statement is used to handle events possible in a particular state of the test component. These events include reception of messages, timer events and termination of other PTCs. The alt statement uses a so-called snapshot logic. However, execution does not stop after a snapshot was taken, so the state of the test component and the queues assigned to it might change in between. So, generally the two most significant factors we consider, while performance evaluating a test component are the matching mechanism and the queuing at the test ports. In our modeling approach, we build an analytic model of test components that use alternatives to handle internal messages and messages coming from the SUT. We describe concurrent queues underlying the test ports of a component with a stochastic process and calculate the steady-state solution. After solving the analytical model, we predict the probability that one of the queues contain a message that is postponed indefinitely, because of the fact that race conditions arise between the queues. The probability of an indefinite postponement is calculated as a function of arrival intensities at the corresponding queues and of other parameters relevant to the implementation of the actual test component.